

WHAT IS CLAIMED IS:

- 1 1. A method comprising:
 - 2 obtaining a location identifier of a destination of a write operation;
 - 3 storing the location identifier in an available entry of an array, said entry being
 - 4 identified by an array pointer having a first value; and
 - 5 changing the value of said array pointer to a second value to identify another
 - 6 available array entry.
- 1 2. The method of claim 1 wherein the entries of the array in which location
- 2 identifiers are stored form a stack of array entries and wherein said storing stores the
- 3 location identifier of a write operation in the next available entry of said array.
- 1 3. The method of claim 2 wherein said, said array pointer is an index, and
- 2 said changing includes incrementing said array pointer index to identify the next entry of
- 3 the array as the next available entry of said array.
- 1 4. The method of claim 3 further comprising:
 - 2 upon completion of a write operation, removing the location identifier of the
 - 3 completed write operation destination from the stack of array entries and reducing the
 - 4 size of said stack of array entries.
- 1 5. The method of claim 4 wherein said reducing includes removing the last
- 2 entry of said stack of array entries by decrementing said array pointer index to identify
- 3 the previous entry of the array as the next available entry of said array.
- 1 6. The method of claim 4 further comprising:
 - 2 upon completion of a write operation, moving the location identifier stored in the
 - 3 last entry of said stack of array entries to the stack entry containing the location identifier
 - 4 of the completed write operation and overwriting the location identifier of the completed
 - 5 write operation with the moved location identifier; and removing the last entry of said
 - 6 stack of array entries by decrementing said array pointer index to identify the previous
 - 7 entry of the array as the next available entry of said array..

1 7. The method of claim 1 further comprising storing the array pointer first
2 value in a data structure entry.

1 8. The method of claim 4 further comprising updating a logical redundancy
2 check of the contents of the reduced stack.

1 9. The method of claim 1 wherein said array is implemented in nonvolatile
2 storage memory.

1 10. A method comprising:
2 obtaining a location identifier of a destination of a write operation;
3 storing the location identifier in an available entry of an array, said entry being
4 identified by an array pointer having a first value;
5 storing the array pointer first value in a data structure entry;
6 changing the value of said array pointer to a second value to identify another
7 available array entry;
8 receiving a data structure pointer upon completion of a write operation;
9 obtaining an array pointer value from the data structure entry identified by the
10 received data structure pointer; and
11 removing the location identifier from the array entry identified by said obtained
12 array pointer value wherein the removed location identifier identifies a completed write
13 operation destination.

1 11. The method of claim 10 wherein the entries of the array in which location
2 identifiers are stored form a stack of array entries and wherein said location identifier
3 storing stores the location identifier of a write operation in the next available entry of said
4 array and wherein said location identifier removing removes the location identifier of the
5 completed write operation destination from the stack of array entries, said method further
6 comprising reducing the size of said stack of array entries.

1 12. A method comprising:
2 obtaining a location identifier of a destination of a write operation;

3 storing the location identifier in an available entry of an array of entries in
4 nonvolatile storage, wherein the entries of the array in which location identifiers are
5 stored form a stack of array entries and wherein said available entry being identified by
6 an index pointer having a first value;

7 incrementing the value of said array pointer to a second value to identify the next
8 entry of the array as the next available entry of said array;

9 receiving a data structure pointer upon completion of a write operation;

10 obtaining an array pointer index value from the data structure entry identified by
11 the received data structure pointer;

12 removing the location identifier from the stack entry identified by said obtained
13 array pointer index value;

14 decrementing the value of said index;

15 comparing said obtained index value to said decremented index value;

16 if said obtained index value is different from said decremented index value,
17 moving a write operation destination location identifier from the stack entry identified by
18 the decremented index value to a stack entry identified by the obtained index value; and

19 updating a logical redundancy check of contents including contents of the stack
20 and the index.

1 13. The method of claim 12 further comprising storing the obtained index
2 value in the data structure entry associated with the location identifier moved to the stack
3 entry identified by the obtained index value.

1 14. An article of manufacture for data storage, wherein the article of
2 manufacture causes operations to be performed, the operations comprising:

3 obtaining a location identifier of a destination of a write operation;

4 storing the location identifier in an available entry of an array, said entry being
5 identified by an array pointer having a first value; and

6 changing the value of said array pointer to a second value to identify another
7 available array entry.

1 15. The article of claim 14 wherein the entries of the array in which location
2 identifiers are stored form a stack of array entries and wherein said storing stores the
3 location identifier of a write operation in the next available entry of said array.

1 16. The article of claim 15 wherein said, said array pointer is an index, and
2 said changing includes incrementing said array pointer index to identify the next entry of
3 the array as the next available entry of said array.

1 17. The article of claim 16 wherein the operations further comprise:
2 upon completion of a write operation, removing the location identifier of the
3 completed write operation destination from the stack of array entries and reducing the
4 size of said stack of array entries.

1 18. The article of claim 17 wherein said reducing includes removing the last
2 entry of said stack of array entries by decrementing said array pointer index to identify
3 the previous entry of the array as the next available entry of said array.

1 19. The article of claim 17 wherein the operations further comprise:
2 upon completion of a write operation, moving the location identifier stored in the
3 last entry of said stack of array entries to the stack entry containing the location identifier
4 of the completed write operation and overwriting the location identifier of the completed
5 write operation with the moved location identifier; and removing the last entry of said
6 stack of array entries by decrementing said array pointer index to identify the previous
7 entry of the array as the next available entry of said array..

1 20. The article of claim 14 wherein the operations further comprise storing the
2 array pointer first value in a data structure entry.

1 21. The article of claim 17 wherein the operations further comprise updating a
2 logical redundancy check of the contents of the reduced stack.

1 22. The article of claim 14 wherein said array is implemented in nonvolatile
2 storage memory.

1 23. An article of manufacture for data storage, wherein the article of manufacture
2 causes operations to be performed, the operations comprising:

3 obtaining a location identifier of a destination of a write operation;
4 storing the location identifier in an available entry of an array, said entry being
5 identified by an array pointer having a first value;
6 storing the array pointer first value in a data structure entry;
7 changing the value of said array pointer to a second value to identify another
8 available array entry;
9 receiving a data structure pointer upon completion of a write operation;
10 obtaining an array pointer value from the data structure entry identified by the
11 received data structure pointer; and
12 removing the location identifier from the array entry identified by said obtained
13 array pointer value wherein the removed location identifier identifies a completed write
14 operation destination.

1 24. The article of claim 23 wherein the entries of the array in which location
2 identifiers are stored form a stack of array entries and wherein said location identifier
3 storing stores the location identifier of a write operation in the next available entry of said
4 array and wherein said location identifier removing removes the location identifier of the
5 completed write operation destination from the stack of array entries, and wherein the
6 operations further comprise reducing the size of said stack of array entries.

1 25. An article of manufacture for data storage, wherein the article of
2 manufacture causes operations to be performed, the operations comprising:
3 obtaining a location identifier of a destination of a write operation;
4 storing the location identifier in an available entry of an array of entries in
5 nonvolatile storage, wherein the entries of the array in which location identifiers are
6 stored form a stack of array entries and wherein said available entry being identified by
7 an index pointer having a first value;
8 incrementing the value of said array pointer to a second value to identify the next
9 entry of the array as the next available entry of said array;
10 receiving a data structure pointer upon completion of a write operation;

11 obtaining an array pointer index value from the data structure entry identified by
12 the received data structure pointer;
13 removing the location identifier from the stack entry identified by said obtained
14 array pointer index value;
15 decrementing the value of said index;
16 comparing said obtained index value to said decremented index value;
17 if said obtained index value is different from said decremented index value,
18 moving a write operation destination location identifier from the stack entry identified by
19 the decremented index value to a stack entry identified by the obtained index value; and
20 updating a logical redundancy check of contents including contents of the stack
21 and the index.

1 26. The article of claim 25 wherein the operations further comprise storing the
2 obtained index value in the data structure entry associated with the location identifier
3 moved to the stack entry identified by the obtained index value.

1 27. A system comprising:
2 a storage device adapted to store an array having an array pointer having a first
3 value which identifies an available entry of said array;
4 a digital data processing apparatus coupled to the storage device, wherein the
5 digital data processing apparatus is configured to:
6 obtain a location identifier of a destination of a write operation;
7 store the location identifier in an available entry of said array identified by
8 said array pointer having said first value; and
9 change the value of said array pointer to a second value to identify another
10 available array entry.

1 28. The system of claim 27 wherein the entries of the array in which location
2 identifiers are stored form a stack of array entries and wherein said storing stores the
3 location identifier of a write operation in the next available entry of said array.

1 29. The system of claim 28 wherein said array pointer is an index, and said
2 changing includes incrementing said array pointer index to identify the next entry of the
3 array as the next available entry of said array.

1 30. The system of claim 29 wherein the digital data processing apparatus is
2 further configured to, upon completion of a write operation, remove the location
3 identifier of the completed write operation destination from the stack of array entries and
4 to reduce the size of said stack of array entries.

1 31. The system of claim 30 wherein said reducing includes removing the last
2 entry of said stack of array entries by decrementing said array pointer index to identify
3 the previous entry of the array as the next available entry of said array.

1 32. The system of claim 30 wherein the digital data processing apparatus is
2 further configured to, upon completion of a write operation, move the location identifier
3 stored in the last entry of said stack of array entries to the stack entry containing the
4 location identifier of the completed write operation and overwrite the location identifier
5 of the completed write operation with the moved location identifier; and remove the last
6 entry of said stack of array entries by decrementing said array pointer index to identify
7 the previous entry of the array as the next available entry of said array..

1 33. The system of claim 27 further comprising a data structure and wherein
2 the digital data processing apparatus is further configured to store the array pointer first
3 value in said data structure entry.

1 34. The system of claim 30 wherein the digital data processing apparatus is
2 further configured to update a logical redundancy check of the contents of the reduced
3 stack.

1 35. The system of claim 27 wherein said storage device is nonvolatile storage
2 memory.

1 36. A storage controller for use with a data storage device and a data structure
2 having entries, comprising:

- 3 a digital data processing apparatus adapted to be coupled to the storage device,
4 wherein the digital data processing apparatus is configured to:
5 store in said storage device an array having an array pointer having a first
6 value which identifies an available entry of said array;
7 obtain a location identifier of a destination of a write operation;
8 store the location identifier in an available entry of said array identified by
9 said array pointer having said first value;
10 store the array pointer first value in a data structure entry;
11 change the value of said array pointer to a second value to identify another
12 available array entry;
13 receive a data structure pointer upon completion of a write operation;
14 obtain an array pointer value from the data structure entry identified by the
15 received data structure pointer; and
16 remove the location identifier from the array entry identified by said
17 obtained array pointer value wherein the removed location identifier identifies a
18 completed write operation destination.
1 37. The storage controller of claim 36 wherein the entries of the array in
2 which location identifiers are stored form a stack of array entries and wherein said
3 location identifier storing stores the location identifier of a write operation in the next
4 available entry of said array and wherein said location identifier removing removes the
5 location identifier of the completed write operation destination from the stack of array
6 entries, and the digital data processing apparatus is further configured to reduce the size
7 of said stack of array entries.
1 38. A storage controller for use with a non-volatile data storage device and a
2 data structure having entries, comprising:
3 a digital data processing apparatus adapted to be coupled to the storage device,
4 wherein the digital data processing apparatus is configured to:

5 store in said storage device an array which includes a stack, said array
6 having an array pointer having a first value which identifies an available entry of
7 said array;
8 obtain a location identifier of a destination of a write operation;
9 store the location identifier in said entry identified by an index pointer
10 having said first value; and
11 increment the value of said array pointer to a second value to identify the
12 next entry of the array as the next available entry of said array wherein said entry
13 identified by said index pointer first value and containing said location identifier
14 becomes a part of the stack;
15 receive a data structure pointer upon completion of a write operation;
16 obtain an array pointer index value from the data structure entry identified
17 by the received data structure pointer;
18 remove the location identifier from the stack entry identified by said
19 obtained array pointer index value;
20 decrement the value of said index;
21 compare said obtained index value to said decremented index value;
22 if said obtained index value is different from said decremented index
23 value, move a write operation destination location identifier from the stack entry
24 identified by the decremented index value to a stack entry identified by the
25 obtained index value; and
26 update a logical redundancy check of contents including contents of the
27 stack and the index.

1 39. The storage controller of claim 38 wherein the digital data processing
2 apparatus is further configured to store the obtained index value in the data structure
3 entry associated with the location identifier moved to the stack entry identified by the
4 obtained index value.